IN THE CLAIMS

Amended claims follow:

1. (Currently Amended) A computer program product embodied on a tangible computer readable medium for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said computer program product comprising:

receiving code to receive at said destination computer operation specifying XML data sent by said source computer;

parsing code to parse said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching code to match each complex data type with an associated execution process available to said destination computer; [[and]]

triggering code to trigger processing by the or each execution process associated with a complex data type within said operation specifying XML data; and

validating said operation specifying XML data received at said destination computer against schema data, where said schema data is sent to said destination computer from said source computer at the same time as said operation specifying XML data;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation.

2. (Previously Presented) A computer program product as claimed in claim 1, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

3. (Original) A computer program product as claimed in claim 1, wherein said operation performed includes making a call to an API available to said destination computer.

4. (Cancelled)

5. (Cancelled)

6. (Cancelled)

7. (Cancelled)

8. (Cancelled)

9. (Cancelled)

10. (Cancelled)

11. (Cancelled)

12. (Previously Presented) A computer program product as claimed in claim 1, wherein said result data is passed from said destination computer to said source computer as XML data.

13. (Original) A computer program product as claimed in claim 1, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

14. (Original) A computer program product as claimed in claim 1, wherein an operation that may be performed by said destination computer includes installing a new execution process.

15. (Original) A computer program product as claimed in claim 1, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

16. (Currently Amended) A computer program product embodied on a tangible computer readable medium for triggering an operation at a destination computer using

data transferred between a source computer and said destination computer, said computer program product comprising:

data forming code to form at said source computer operation specifying XML data containing one or more complex data types; and

transmitting code to transmit from said source computer to said destination computer said operation specifying XML data;

wherein the or each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation;

wherein schema data is transmitted from said source computer to said destination computer at the same time as said operation specifying XML data.

17. (Previously Presented) A computer program product as claimed in claim 16, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

18. (Original) A computer program product as claimed in claim 16, wherein said operation performed includes making a call to an API available to said destination computer.

19. (Cancelled)

20. (Cancelled)

21. (Cancelled)

22. (Cancelled)

23. (Cancelled)

24. (Cancelled)

25. (Cancelled)

26. (Cancelled)

27. (Previously Presented) A computer program product as claimed in claim 16, wherein said result data is passed from said destination computer to said source computer as XML data.

28. (Original) A computer program product as claimed in claim 16, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

[[3]]29. (Currently Amended) computer program product as claimed in claim 16, wherein an operation that may be performed by said destination computer includes installing a new execution process.

30. (Original) A computer program product as claimed in claim 16, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

31. (Currently Amended) A method of triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said method comprising the steps of:

receiving at said destination computer operation specifying XML data sent by said source computer;

parsing said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching the or each complex data type with an associated execution process available to said destination computer; [[and]]

triggering processing by the or each execution process associated with a complex data type within said operation specifying XML data; and

validating said operation specifying XML data received at said destination computer against schema data, where said schema data is sent to said destination computer from said source computer at the same time as said operation specifying XML data;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation.

32. (Previously Presented) A method as claimed in claim 31, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

33. (Original) A method as claimed in claim 31 wherein said operation performed includes making a call to an API available to said destination computer.

34. (Cancelled)

35. (Cancelled)

36. (Cancelled)

37. (Cancelled)

38. (Cancelled)

39. (Cancelled)

40. (Cancelled)

41. (Cancelled)

42. (Previously Presented) A method as claimed in claim 31, wherein said result data is passed from said destination computer to said source computer as XML data.

43. (Previously Presented) A method as claimed in claim 31, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

44. (Previously Presented) A method as claimed in claim 31, wherein an operation that may be performed by said destination computer includes installing a new execution process.

45. (Previously Presented) A method as claimed in claim 31, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

46. (Currently Amended) A method of triggering an operation at a destination computer using data
transferred between a source computer and said destination computer, said method comprising the steps of:

forming at said source computer operation specifying XML data containing one or more complex data types; and

transmitting from said source computer to said destination computer said operation specifying XML data;

wherein the or each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation;

wherein schema data is transmitted from said source computer to said destination computer at the same time as said operation specifying XML data.

47. (Previously Presented) A method as claimed in claim 46, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

48. (Original) A method as claimed in claim 46, wherein said operation performed includes making a call to an API available to said destination computer.

49. (Cancelled)

50. (Cancelled)

51. (Cancelled)

52. (Cancelled)

53. (Cancelled)

54. (Cancelled)

55. (Cancelled)

56. (Cancelled)

57. (Previously Presented) A method as claimed in claim 46, wherein said result data is passed from said destination computer to said source computer as XML data.

58. (Original) A method as claimed in claim 46, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

59. (Original) A method as claimed in claim 46, wherein an operation that may be performed by said destination computer includes installing a new execution process.

60. (Original) A method as claimed in claim 46, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

61. (Currently Amended) Apparatus for triggering an operation at a destination computer using data
transferred between a source computer and said destination computer, said apparatus comprising:

receiving logic to receive at said destination computer operation specifying XML data sent by said source computer;

parsing logic to parse said operation specifying XML data to identify one or more complex data types within said operation specifying XML data;

matching logic to match the or each complex data type with an associated execution process available to said destination computer; [[and]]

triggering logic to trigger processing by the or each execution process associated with a complex data type within said operation specifying XML data; and

validating said operation specifying XML data received at said destination computer against schema data, where said schema data is sent to said destination computer from said source computer at the same time as said operation specifying XML data;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation.

62. (Previously Presented) Apparatus as claimed in claim 61, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

63. (Original) Apparatus as claimed in claim 61, wherein said operation performed includes making a call to an API available to said destination computer.

64. (Cancelled)

65. (Cancelled)

66. (Cancelled)

67. (Cancelled)

68. (Cancelled)

69. (Cancelled)

70. (Cancelled)

71. (Cancelled)

72. (Previously Presented) Apparatus as claimed in claim 61, wherein said result data is passed from said destination computer to said source computer as XML data.

73. (Original) Apparatus as claimed in claim 61, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

74. (Original) Apparatus as claimed in claim 61, wherein an operation that may be performed by said destination computer includes installing a new execution process.

75. (Original) Apparatus as claimed in claim 61, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

76. (Currently Amended) Apparatus for triggering an operation at a destination computer using data transferred between a source computer and said destination computer, said apparatus comprising:

data forming logic to form at said source computer operation specifying XML data containing one or more complex data types;

transmitting logic to transmit from said source computer to said destination computer said operation specifying XML data;

wherein the or each complex data type within said operation specifying XML data corresponds to an execution process available to said destination computer to be triggered to operate;

wherein said operation performed includes configuring said destination computer to execute a computer program;

wherein said execution process maps configuration data specified within said operation specifying XML data to a configuration data store of said destination computer;

wherein said configuration data store is one of:

a Windows Registry entry;

an INI file;

a DAPI store; and

a database entry;

wherein an identifier of said execution process within said complex data type includes at least one of:

data specifying a computer file to trigger said execution process;

data specifying a communication channel to trigger said execution process; and

data specifying an operating system command to trigger said execution process;

wherein said operation includes returning result data from said destination computer to said source computer in dependence upon said operation performed by said execution process;

wherein said result data includes data specifying existing configuration data of said destination computer;

wherein said execution process maps existing configuration data of said destination computer stored within said configuration data store of said destination computer to said result data to be returned to said source computer;

wherein said operation specifying XML data is parsed after validating said operation specifying XML data to extract at least one identifier for mapping said at least one identifier to an available execution process;

wherein said operation specifying XML data includes parameter data used by said execution process in said operation;

wherein schema data is transmitted from said source computer to said destination computer at the same time as said operation specifying XML data.

77. (Previously Presented) Apparatus as claimed in claim 76, wherein parameter data used by said execution process is represented by data within said complex data type of said execution process.

78. (Original) Apparatus as claimed in claim 76, wherein said operation performed includes making a call to an API available to said destination computer.

79. (Cancelled)

80. (Cancelled)

81. (Cancelled)

82. (Cancelled)

83. (Cancelled)

84. (Cancelled)

85. (Cancelled)

86. (Cancelled)

87. (Previously Presented) Apparatus as claimed in claim 76, wherein said result data is passed from said destination computer to said source computer as XML data.

88. (Original) Apparatus as claimed in claim 76, wherein said operation includes returning result data from said destination computer to said source computer in dependence upon whether or not said execution process is available to said destination computer.

89. (Original) Apparatus as claimed in claim 76, wherein an operation that may be performed by said destination computer includes installing a new execution process.

90. (Original) Apparatus as claimed in claim 76, wherein said operation specifying data is validated by said destination computer by comparing with a template defining valid data.

91. (Cancelled)

92. (Currently Amended) A computer program product as claimed in claim 1, further comprising validating said operation specifying XML data received at said destination computer against said schema data, where said schema data is present in said destination computer when said operation specifying XML data is sent.

93. (Cancelled)

94. (Cancelled)

95. (New) A computer program product as claimed in claim 1, wherein said validating of said operation specifying XML data and said schema data transmitted from said source computer to said destination computer at the same time generates a validation result.

96. (New) A computer program product as claimed in claim 95, wherein said validation result triggers at least one of a valid configuration response and an invalid configuration response.

97. (New) A computer program product as claimed in claim 96, wherein said invalid configuration response generates an error message.

98. (New) A computer program product as claimed in claim 96, wherein said valid configuration response starts execution of an associated computer program.